Tle

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/762,794	01/22/2004	Stefan Daume	10125/00101	5381

7590 07/02/2007

FAY KAPLUN & MARCIN, LLP
Suite 702
150 Broadway
New York, NY 10038

EXAMINER	
VU, TUAN A	
ART UNIT	PAPER NUMBER
2193	
MAIL DATE	DELIVERY MODE
07/02/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	**Application No.** 10/762,794	**Applicant(s)** DAUME ET AL.
	Examiner Tuan A. Vu	**Art Unit** 2193

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *01 May 2007*.

2a)☒ This action is **FINAL**. 2b)□ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-10 and 13-27* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-10 and 13-27* is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)□ The drawing(s) filed on _____ is/are: a)□ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) □ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) □ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application

6) □ Other: _____.

DETAILED ACTION

1. This action is responsive to the Applicant's response filed 5/1/07.

As indicated in Applicant's response, claims 1, 15 have been amended, claim 27 added, and claims 11-12 canceled. Claims 1-10, 13-27 are pending in the office action.

Claim Rejections - 35 USC § 103

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. Claims 1-10, 13-27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Ankireddipally et al., USPN: 6,971,096 (hereinafter Anki_dipally) and further in view of Lai, USPubN: 2005/0044197 (hereinafter Lai).

As per claim 1, Anki_dipally discloses a method, comprising the steps of:

generating a usage task (e.g. Transaction request 284 - Fig. 2; data structure – col. 16, lines 22-26) from (e.g. CXC 34 – Fig. 2; application 34, messages - col. 16, lines 11-22 – Note: *message type*, *acknowledge data*, *transaction log*, *protocol data* related to asynchronous messages – user session–related communication - received by the CXC application read on usage data – see col. 10, line 66 to col. 11, line 9);

constructing a pattern graph (e.g. *transaction definition ... conforms DAG* - col. 14, line 60 to col. 15, line 14) from the usage task;

constructing a model graph which represents a space of equivalents to the usage task (e.g. Transaction instance 270 – Fig. 3; transaction instance uses transaction definition 282 ... to produce … structure 270 - col. 16, line 41-48) represented by the pattern graph (e.g. transaction definition 280 – col. 16, line 21-40); and

extracting sub-graphs from the model graph (e.g. col. 15, line 15-22; transaction definition 280 – col. 16, line 21-40), wherein each of the extracted sub-graphs is isomorphic to the pattern graph (Note: extracting XML conformant transaction definition of in terms objects, name – see Table 1, Table 2, col. 19, 21 -- according to the graph and DTD requirement of the request to yield sub-graphs of the DAG 270 – Fig. 3 – reads on subgraphs thereof being isomorphic to pattern graph – see XML col. 15, line 1-62);

creating virtual tasks from each of the extracted sub-graphs (template col. 35, line 29 to col. 56; Table 9, col. 35-36 – Note: virtual tasks is represented by OPID and OPLINK and definition to join sub-definition of tasks --or sub-graphs --concerning a virtual task implemented via virtual state from template-based declaration)

Anki_dipally does not explicitly disclose performing a stress-test on an application program using the virtual tasks. Using XML structure or messaging to address web services and database query transaction using a model of datastore of XML definition analogous to Anki_dipally (see Anki_dipally: Fig. 1-2), Lai discloses scalability of the services to be rendered for the network request and applications as well as security therein (see Fig. 91, 97) and also teaches stress test (para 0481, pg. 23). Based on Anki_dipally concern for complex distribution of multi-threaded request (Threading Models - col. 12-13 and the test for a given application as set forth above, the desirability to maintain service and test to verify of its proper operation is suggested. Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to additionally implement the testing by Anki_dipally with a possibility of a stress-test as mentioned by Lai because this feature would impart more knowledge to the extend of the services by Kai in way that preventive adjustment over the multi-threaded paradigm can be effectuated for the service to be maintained with the purpose to handle the scalibilty desired in multi-threaded handling as set forth above and in view of Kai's teaching..

As per claims 2-3, Anki_dipally discloses capturing the usage data from a user session; wherein the capturing step includes one of retrieving a log file of an application program and intercepting user calls to the application program (e.g. *tracking transaction threads* – col. 11, lines 38-43; Fig. 2 – Note: retrieved incoming packets based on transaction threads received at service 200 – Fig. 2 – reads on retrieving calls from a logging structure of stored packets related to threads or messages received – see col. 12, lines 62 to col. 13, line 25).

As per claim 4, Anki_dipally discloses that usage data includes API calls (Note: a application user's request to effect queried data from a database – see Fig. 1-2 -- reads on API calls -- see request messaging *40* – Fig 1 – wherein XML content represented via API calls within a DOM also reads on API invocation – see col. 12, line 3-16).

As per claim 5, Anki_dipally discloses wherein the generating task includes

separating the usage data into generalizable entities and non-generalizable entities, wherein the pattern graph is constructed using only the generalizable entities (e.g. col. 14, line 55 to col. 15 line 13 – Note: transaction definition as XML form reads on data being generalized as reuseable metadata – see col. 17, lines 32-34; Example 10-13, col. 21-29)

As per claim 6, Anki_dipally discloses wherein the pattern graph construction step includes the sub-step of creating a fully qualified name for each of the generalizable entities, the fully qualified name including a local entity name and an associative qualifier (see Example 10-

13, col. 21-29; TABLE 2, 4-5, 7, col. 21-30 – Note: tag name enclosing tag body of XML or DTD format with header tag and indentation thereof reads on local entity name and its associative qualifier – see Fig. 5).

As per claims 7-8, Anki_dipally discloses wherein the pattern graph construction step further includes the sub-steps of merging each fully qualified name into a node of the pattern graph (DAG – col. 15, lines 6-13); and merging each of the associative qualifiers into an edge of the pattern graph (refer to claim 6 – Note: tagged variable reads on node of DAG and indentation of tag header reads on link to subgraph -- see Fig. 5); wherein the pattern graph is a directed acyclic graph (DAG – col. 15, lines 6-13).

As per claim 9-10, Anki_dipally discloses wherein the model graph (e.g. Transaction instance 270; col. 18 lines 1-19; Fig. 4-5) is a directed acyclic graph; wherein the model graph retains all the dependencies of the pattern graph (e.g. col. 16 lines 32-61; col. 17, lines 9-11).

As per claim 13, Anki_dipally discloses determining an equivalent model graph node by matching node properties of a pattern graph node to node properties of a model graph node (re claim 1: extracting XML conformant transaction definition of in terms objects, name – see Table 1, Table 2, col. 19, 21; col. 16 lines 32-61; col. 17, lines 9-11 -- according to the graph and DTD requirement of the request to yield sub-graphs of the DAG 270 – Fig. 3 – reads on subgraphs thereof being isomorphic to pattern graph – see XML col. 15, line 1-62, hence pattern graph node conforming with model graph node as per the DAG approach).

As per claim 14, Anki_dipally discloses extraction of corresponding sub-graphs in the definition structures of a XML DAG with creation of a transaction instance model (col. 16 lines 32-61; col. 17, lines 9-11) but does not explicitly disclose the mapping or extraction step includes the sub-step of: performing one of a breadth-first search(BFS) and a depth-first search (DFS) of the model graph. A tree as taught by XML or DTD in a DOM context implies a form of scanning and traversal effectuated either by depth first or breath first; and this can be inferred from the parser (e.g. DOM - col. 32, line 50 to col. 33 line 15). It would have been obvious for one skill in the art to traverse the markup hierarchy of the transaction definition DAG by Anki_dipally so that either such traversal be DFS or BFS because each of such method would yield a desired pattern of knowledge required to build a record of operations information, requirements or variables/parameters (see Anki_dipally: Example 10-13, col. 21-29; TABLE 2, 4-5, 7, col. 21-30) sufficient to support the creation of the transaction instance by Anki_dipally as set forth above.

As per claim 15, Anki_dipally discloses a system, comprising:

usage task generation module configured to *generate usage task from usage data*;

a pattern graph construction module configured to *construct a pattern graph from a usage task*;

a model graph construction module configured to *construct a model graph which represents a space of equivalents to the usage task represented by the pattern graph*; and

an extraction module configured to *extract sub-graphs from the model graph, wherein each of the extracted sub-graphs is isomorphic to the pattern graph*;

virtual tasks creation module to *create virtual tasks from each of the extracted sub-graphs* (template col. 35, line 29 to col. 56; Table 9, col. 35-36 – Note: virtual tasks is represented by OPID and OPLINK and definition to join sub-definition of tasks --or sub-graphs --concerning a virtual task implemented via virtual state from template-based declaration) all of which limitations (italicized text) having been addressed in claim 1 above.

Anki_dipally does not explicitly disclose a stress-test performance module to perform stress-test on an application program using the virtual tasks but this limitation has been addressed in claim 1.

As per claims 16-17, Anki_dipally discloses wherein the pattern graph includes nodes and edges (col. 14, line 55 to col. 15 line 13 – Note: transaction definition as XML and DAG form reads on node and edges – see col. 17, lines 32-34; Example 10-13, col. 21-29) wherein each of the nodes includes a node label, the node label including a set of node properties for each of the nodes (see Table 1, Table 2, col. 19, 21; col. 16 lines 32-61; col. 17, lines 9-11).

As per claims 18-19, Anki_dipally discloses wherein the node label is null (see XML/COM …parser - col. 11/12 --Note: parsing language having a pointer to traverse a tree necessarily envisions a initial null value thereto to prevent crash when no data is found at the end of a referenced position); wherein each of the nodes includes a node value (see Fig. 5-6).

As per claim 20, Anki_dipally discloses wherein each of the edges is directed to reflect the ordering of the nodes (refer to claim 6 - Note: tag name enclosing tag body of XML or DTD format with header tag and indentation thereof reads on local entity name and its associative qualifier – see Fig. 5).

As per claims 21-22, Anki_dipally discloses wherein the model graph also includes nodes and edges, wherein each of the nodes of the model graph is equivalent to at least one of the nodes of the pattern graph and each of the edges of the model graph is equivalent to at least one of the edges of the pattern graph; wherein an equivalent model graph node is determined by matching node properties of one of the pattern graph nodes (Note: the subject matter herein has been addressed in the corresponding rejection set forth in claim 13 -- Note: extracting XML conformant transaction definition of in terms objects, name – see Table 1, Table 2, col. 19, 21 -- according to the graph and DTD requirement of the request to yield sub-graphs of the DAG 270 – Fig. 3 – reads on subgraphs thereof being isomorphic to pattern graph – see XML col. 15, line 1-62).

As per claims 23-25, refer the rejections of claims 8-10, respectively.

As per claim 26, Anki_dipally discloses a computer-readable storage medium storing a set of instructions, the set of instructions capable of being executed by a processor, the set of instructions performing the steps of:

generating a usage task from usage data;

constructing a pattern graph from the usage task;

constructing a model graph which represents a space of equivalents to the usage task represented by the pattern graph; and

extracting sub-graphs from the model graph, wherein each of the extracted sub-graphs is isomorphic to the pattern graph;

all of which step limitations having been addressed in claim 1 or 15.

As per claim 27, Anki_dipally discloses creating virtual tasks from each of the extracted sub-graphs (template col. 35, line 29 to col. 56; Table 9, col. 35-36 – Note: virtual tasks is represented by OPID and OPLINK and definition to join sub-definition of tasks --or sub-graphs --concerning a virtual task implemented via virtual state from template-based declaration); but Anki_dipally does not explicitly disclose performing a stress-test on an application program using the virtual tasks; however, this limitation has been rejected as obvious from above.

Response to Arguments

4. Applicant's arguments filed 5/1/07 have been fully considered but they are not persuasive. Following are the Examiner's observation in regard thereto.

USC 35 § 102(e) Rejection:

(A) In light of the amendments, the arguments in accordance with the § 102 ground of rejection are mostly **moot** in view of the new grounds of rejection. Nevertheless, in response to Applicants' argument that Anki_dipally fails to fails to teach or suggest using 'usage data to generate usage tasks', and rather teaches instead generate instances from definition (Appl. Rmrks pg. 8, top para), following is the Examiner's analysis. *Usage data* are viewed as data related to user during a session; and because the NW session encompasses messages and requests, the identification data from the messages being captured, stored, parsed (e.g. for identifying their nature, ID, type), then addressed (e.g. by a CXC application) in accordance to protocol of NW communication as taught by Anki_dipally (re claim 1: col. 10, line 66 to col. 11, line 9), amounts to what is recited as 'usage data'. Further, the constructs of a data being put together in response to addressing the user's message amount to what is recited as 'usage task', the representation of what needs to be done when all the content of the messages are addressed (e.g. data structure – col. 16, lines 22-26). Usage data can encompass data pertinent to a user, or a user-specific session, or just a request from that user; all of which having been cited or mapped in the rejection. The language of the claim in terms of detailing how the limitation is all about appears insufficient for the teachings by Anki_dipally to be precluded from the above mapping. Applicant's arguments fail to comply with 37 CFR 1.111(b) because they amount to a general allegation that the claims define a patentable invention without specifically pointing out how the language of the claims patentably distinguishes them from the references.

(B) Claims 15 and 26 recite the language being analyzed above concerning the usage data and usage task. The arguments therefor (Appl. Rmrks, pg. 9) would be referred to section A.

As a result of the above, the claims stand rejected as set forth in the office Action.

Conclusion

5. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Tuan A Vu whose telephone number is (272) 272-3735. The examiner can normally be reached on 8AM-4:30PM/Mon-Fri.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai An can be reached on (571)272-3756.

The fax phone number for the organization where this application or proceeding is assigned is (571) 273-3735 (for non-official correspondence - please consult Examiner before using) or 571-273-8300 (for official correspondence) or redirected to customer service at 571-272-3609.

Any inquiry of a general nature or relating to the status of this application should be directed to the TC 2100 Group receptionist: 571-272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Tuan A Vu
Patent Examiner,
Art Unit 2193
June 26, 2007